

Application #10/035,905
Amendment dated October 19, 2005

Remarks:

In the Office Action mailed on May 19, 2005, the Examiner rejected claims 1-27. Applicant amends claims 1, 3, 11, 13, 21, and 23 herein. Claims 1-27 are pending in the application.

Specification

Applicants have amended page 6 of the Specification to more clearly describe the invention. The change is supported by the Specification as originally filed. Accordingly, no new matter has been added.

35 USC 102(b)

Claims 1-4, 11-14, and 21-24 were rejected under 35 USC 102(b) as anticipated by Cable, US Pat. No. 5,999,728 (hereinafter "Cable"). Applicants have amended the independent claims and several of the dependent claims to more clearly define the invention. To the extent that the Examiner believes that the anticipation rejection with respect to Cable applies to the claims as amended, Applicants traverse.

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. The identical invention must be shown in as much detail as is contained in the claim." MPEP 2131. That standard cannot be met using the Cable reference.

Cable addresses an entirely different problem from the problem solved by the applicants. It is therefore not surprising that Cable does not teach or suggest applicants' novel and non-obvious invention as recited in the claims.

A summary of Cable may aid the analysis under 35 USC 102(b). Cable provides a system for porting a toolkit of a graphical user interface from one window-based platform to another window-based platform.

Page 9 of 15

A49-1 Amend & Resp v10.doc

Application #10/035,905
Amendment dated October 19, 2005

Cable's method addresses the problem of portability that arises "because the interactions between objects are defined by a model-view-controller developed as a platform specific implementation for use with a specific window-based environment, the toolkit cannot be readily ported from one window-based platform to another." Cable, Col. 2, Lines 36-40.

Furthermore, Cable states "[t]he amount of computer code which must be revised to use a toolkit defined for one window-based platform with another window-based platform is therefore excessive and impractical." Cable, Col. 2, lines 52-55. Therefore, Cable states, "it would be desirable to abstract the events received from a window-based system so that the primary portions of code used to represent the toolkit of an object oriented graphical user interface can remain unchanged, and therefore be easily ported to any of a variety of window based platforms." Cable, Col.2, line 65-Col. 3, line 3.

As part of the solution for the portability of a windows-based graphic user interface toolkit, Cable translates native notifications implemented with respect to a first window-based platform into abstracted notifications for use in defining the graphical user interface. "As a result, each object of the graphical user interface toolkit can be registered with an abstracted notification received from any of plural window-based platforms. Cable, Col. 4, lines 30-40. As Cable deals with graphical user interfaces, naturally, "the notifications can correspond an event such as a key stroke, activation of a push button, an iconified window, and so forth". Cable, Col. 4, lines 41-43. "The functional signature used to represent an abstract notification constitutes a behavioral specification ... to which model, view, and controller object implementations can be made to conform". Col. 4, lines 48-51.

Applicants address an entirely different problem. "In one embodiment of the present invention, changes in the observable state of the system may be utilized to consolidate transactions and commit the

Application #10/035,905
Amendment dated October 19, 2005

applicable data of the consolidated transactions together because the client will not rely upon such data until a change in the observable state occurs." Specification, Page 6, Lines 1-4. In other words, Applicants present a solution in which the number of writes to a non-volatile memory is minimized by avoiding writes to that memory at the conclusion of transactions other than when a change in observable state could have occurred. It should be noted that in the context of the present invention a transaction is a sequence of operations that must either complete successfully or that may be rolled back to the status quo ante. Similarly, a change in observable state, in the context of the invention, occurs when the current state of the system would be available to those accessing the system from the outside world.

As Applicants observe in the Background:

"Many data processing systems (also referred to herein as "systems") require transactional operations to function successfully. A transactional operation is a set of instructions that must (as a set) succeed completely. If a failure (computational or otherwise) occurs before the set of instructions has succeeded completely, the system's values must be restored to the state they were in before the transaction was attempted." Specification, Page 1, lines 12-17.

To ensure the "transactional" characteristic of a series of operations, the beginning of the series of operations is marked, the individual operations performed, and at the end the results from the operations are committed (i.e., the system state is updated). If the transactions fail, the state of the system can be rolled back to the state at the beginning of the sequence of operations that make up a transaction.

"Generally speaking, the invention contemplates optimizing the processing of transactions within a data processing system by minimizing the number of commits required for transactions that have successfully completed. In systems (such as smart cards) utilizing persistent, non-

Application #10/035,905
Amendment dated October 19, 2005

volatile memory such as EEPROM, the invention may (as a result of minimizing the number of commits) increase the operational life of such persistent, non-volatile memory. The invention utilizes a change in the observable state of the system performing the transactions in order to consolidate two or more transactions that have completed successfully and commit the resulting data of the consolidated transactions together rather than committing the resulting data of each transaction separately." Specification, Page 5, lines 10-18.

"A person skilled in the art will appreciate that changes in observable state generally include any means by which the state of the system is made available to the outside world in such a way that those accessing the services of the system could reasonably become aware of the state." Specification, page 5, lines 20-23. "From the viewpoint of the client, the observable state of the system may be defined to generally include only that part of the externally visible state of the system as perceived by the client." Specification, page 5, lines 31-34.

From the foregoing, it should be apparent that Cable and Applicants address entirely different problems. Therefore, it should not come as a surprise, that Cable does not teach or suggest the invention claimed by Applicants. Applicants claim, for example, "examining the computer code being executed for a transaction after the execution of which a change in observable state could occur". Cable does not teach or suggest such a limitation.

The Examiner states in the office action that "Cable discloses in column 4 lines 43+ that state changes in a window's based platform are encapsulated in abstract notifications and, in column 6 lines 59+ that the abstract notifications can be stored in a readable memory medium." Office Action, Page 2, numbered paragraph 2. It should be noted, however, that Cable's state changes have no relationship to transactions, wherein a

Application #10/035,905
Amendment dated October 19, 2005

transaction is a sequence of operations that must either complete correctly or that may be rolled back to a prior state.

Applicants further claim (in claim 1) that “storing data for ... a sequence ... of transactions between a [first] location ... and the transaction after the execution of which a change in observable state would occur, wherein the data is stored within the computer code” and “responsive to detecting a change in observable state, committing a portion of the stored data to non-volatile memory.” In other words, the data associated with a sequence of transactions – again please consider that a transaction has the specific meaning of a series of operations that either must complete correctly or be rolled back – is stored within the computer code until an observable state may have occurred. Intermediate data is not committed to the non-volatile memory.

The Examiner has pointed to the following passage in Cable: “a computer readable storage medium 312 for storing data, such as the abstractions of the notifications from the native window-based platform and/or, a table for mapping events from the target window-based platform to the various abstracted notifications.” While the Applicants make no claim to storing data per se, it should be noted that the claim limitation sets forth that data is stored in non-volatile memory in response to a potential change in observable state and not for transactions preceding that event. A general teaching in a reference that some data may be stored in a computer readable storage medium is not specific enough to constitute a teaching of the specific limitations recited in Applicants novel and non-obvious claim.

For these reasons, Cable does not teach or suggest each and every element set forth in the claim; much less “in as much detail as is contained in the claim”. Accordingly, Claim 1 is not anticipated by Cable and should be allowed. Claims 11 and 21 recite analogous limitations and are

Page 13 of 15

A49-1 Amend & Resp v10.doc

Application #10/035,905
Amendment dated October 19, 2005

therefore patentable over Cable for the same reasons given in support of Claim 1.

Claims 3-4, 12-14, and 22-24 depend from Claims 1, 11, and 21, respectively, incorporate the limitations of their respective base claims, set forth further unique and non-obvious combinations, and are, therefore, patentable for the reasons given in support of Claims 1, 11, and 21 and by virtue of such further combinations.

35 USC 103

Claims 5-10, 15-20, and 25-27 were rejected under 35 USC 103(a) as unpatentable over Cable in view of "well-known prior art". Applicants traverse.

As discussed hereinabove in response to the rejection under 35 USC 102, Cable does not teach or suggest the invention as set forth in Claims 1, 11, and 21. Claims 5-10, 15-20, and 25-27 depend from Claims 1, 11, and 21, respectively, incorporate the limitations of their respective base claims, set forth further unique and non-obvious combinations, and are, therefore, patentable for the reasons given in support of Claims 1, 11, and 21 and by virtue of such further combinations.

The application is now deemed to be in condition for allowance and notice to that effect is solicited.

CONCLUSION

It is submitted that all of the claims now in the application are allowable. Applicants respectfully request consideration of the application and claims and its early allowance. If the Examiner believes that the prosecution of the application would be facilitated by a telephonic

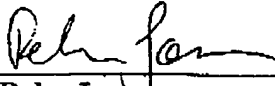
Application #10/035,905
Amendment dated October 19, 2005

interview, Applicants invite the Examiner to contact the undersigned at the number given below.

Applicants respectfully request that a timely Notice of Allowance be issued in this application.

Respectfully submitted,

Date: 10/19/2005


Pehr Jansson
Registration No. 35,759

Anderson & Jansson, LLP
9501 N. Capital of TX Hwy.
Suite 202
Austin, TX 78759
512-750-3046
678-868-0101 (Fax)
pehr@anjanlaw.com